

# Emergency Alarm

Martin Tměj

Faculty of Informatics and Management  
University of Hradec Kralove,  
Hradec Kralove, Czech Republic  
martin.tmej@uhk.cz

**Abstract—** Mobilní telefon je všudypřítomný, nosíme ho stále u sebe a neuděláme bez něj už ani krok. Proto je snaha dostat co nejvíce informací právě na mobilní telefony. Počasí, aktuální dění ve světě, sport a jiné. Tento článek se zabývá problémem získávání informací o mimořádných událostech a upozornění na něj pomocí telefonu. Byla vytvořena aplikace, která získává informace o počasí, konkrétně o teplotě a vlhkosti vzduchu a na základě toho zjišťuje, zda je možný vznik námrazy, nebo náledí. Aplikace tuto informaci sděluje uživateli přes jeho nastavený budík, pokud teplotní podmínky v jeho lokalitě naznačují vznik námrazy, nebo náledí je posunut budík o 15 minut. To umožňuje například uživateli očistit si auto nebo vyjít dříve aby stihl své ranní aktivity včas.

**Keywords-** Budík, Notifikace, Mimořádné události, Námraza, Windows Phone

## I. INTRODUCTION/ÚVOD

Dnes v 21. století si už nikdo nedokáže představit, že by se měl vzdát moderních prvků civilizace. Během posledních pár let se stala všelijaká elektronická zařízení součástí našeho života. Už si mnozí z nás bez nich nedokáží život ani představit. Typickým zástupcem takového zařízení je mobilní telefon, který nám slouží ne jenom jako přenosný telefon ale i jako kalendář, budík, fotoaparát, internetový prohlížeč, hudební přehrávač a mnoho dalších[1]. Z toho důvodu že ho nosíme stále u sebe a vyžadujeme od něj také stále více funkcí. Jednou z těchto funkcí je upozornění na nějaké události, nejdříve to bylo obyčejné buzení, poté přibýlo upozornění na schůzky, svátky, narozeniny a nepřeberné množství dalších událostí, na které chceme být upozorňováni[2]. Tím se dostáváme k problému, kterým se zabývá tento článek. Jak upozornit uživatele na mimořádné události např. náledí, povodně, požár atd.

V rámci článku bude vytvořena aplikace na platformě Windows Phone 8 z toho důvodu zde budou popsány metody upozornění, které se právě vyskytují na této platformě.

Způsob, kterým bude uživatel informován o mimořádných událostech, byl zvolen formou notifikací na mobilním telefonu. Jsou tři způsoby jak upozornit uživatele že si aplikace žádá jeho pozornost[3]. První způsob je Live Tiles ("Živá dlaždice"), která je umístěna na hlavní obrazovce. Další je Secondary Tiles ("vedlejší dlaždice") a

poslední způsob je Notifikací. Tímto způsobem upozornění uživatele se bude právě tento článek zabývat.

Živá dlaždice má za úkol nalákat uživatele k otevření aplikace. Samotné dlaždice jsou vstupní bránou do aplikace, pomocí níž může uživatel snadno otevřít požadovanou aplikaci. Pokud je dlaždice nastavena jako "živá" může na sobě zobrazit doplňující informace a tak uživatele upozorňovat na nové události v aplikaci.

Vedlejší dlaždice umožňuje, aby uživatel v aplikaci označil funkci a tu poté přesunul na hlavní obrazovku v podobě vedlejší dlaždice, nově vytvořená dlaždice poté slouží jako odkaz na zvolenou funkci v aplikaci. Jako klasický příklad může být aplikace na počasí, kde si uživatel může vytvořit vedlejší dlaždici pro každé město a sledovat tak v těchto městech počasí aniž by je musel opětovně vyhledávat v aplikaci. Díky této funkci si může každý uživatel personalizovat svoji aplikaci.

Posledním zástupcem je toast notifikace ("vyskakovací upozornění"). Toto upozornění je skvělou cestou jak sdělit uživateli, že se něco děje. Tato notifikace vyskočí na horní částí displeje, tím překryje aktuální aplikaci, a vydá zvuk. Tímto způsobem je uživatel bezpečně upozorněn a je nelehké toto upozornění přehlédnout.

Notifikace lze dělit i jiným způsobem, podle mechanismu spuštění[4].

- Scheduled Notification - notifikace, která je spuštěna v určitý čas a určité datum na základě naplánování od uživatele.
- Period Notification - pravidelné upozornění které se stále opakuje, tato notifikace je hojně využívána u živé dlaždice. Kde se například bude každou hodinu zobrazovat aktuální počet nepřečtených emailů.
- Push Notification - umožňuje zaslat kdykoli notifikaci z vnějšího serveru, pokud je telefon zapnut. Tento druh notifikace se používá pro odběry novinek ze serverů. Například chceme upozorňovat na to, když vyjde do kin námi očekávaný film, nebo se zveřejní článek, na který jsme čekali. Tato notifikace je zaslána ze serveru a tak nevyžaduje, aby se aplikace pravidelně někam připojovala jak je tomu u Periodic notifikace.

Ukázková aplikace bude kombinovat první (Scheduled) a poslední (Push) způsob spuštění notifikace. Jak tyto informace zaslat do mobilního telefonu bude pojednávat následující kapitola.

## II. PROBLEM DEFINITION/ DEFINICE PROBLÉMU

Jak bylo popsáno v předešlé kapitole téma, kterým se článek bude zabývat, je informování uživatele o mimořádných událostech pomocí upozornění na mobilním telefonu. Informace o událostech by měli být získané z nějakého volně dostupného zdroje. Tím jsou zaručeny nízké náklady na aplikaci a lze tedy předpokládat její rychlé rozšíření.

Aplikace by měla využívat internetové připojení pro získání dat z webových služeb, v ideální případě přes Wifi. Tím bude zaručena i další funkce aplikace a tím je lokalizace uživatele. Pokud bude známa lokace uživatele, můžeme vyhledávat informace, které se dějí v jeho okolí. V případě, že v okolí nebude žádné Wifi připojení, uživatel se může připojit i přes mobilní internet, v tomto případě se o lokaci postará GPS. Pro úsporu baterie telefonu by bylo dobré, aby se poloha uživatele vyhledala pouze jednou a to při spuštění aplikace. Podobné pravidlo by mělo platit i pro internetové připojení, kontrola mimořádných událostí by měla být volitelná uživatelem a to po jedné hodině. Menší časový úsek by mohl rychle vybit baterii.

Podobný problém řešili na Columbijské univerzitě v roce 2001 v USA [5]. Vláda poskytovala efektivní systém varování obyvatelstva před mimořádnými událostmi. Na univerzitě se zabývali myšlenkou jak tento systém varování zpřístupnit mobilním telefonům. Využili k tomu SIP protokol (Session Initiation Protocol) což je signalizační protokol aplikační vrstvy, který se používá pro Internetové hovory (VOIP), instant messaging (psaní na internetu s přáteli) a upozornění na různé události. Pomocí tohoto protokolu mohou uživatelé odesílat požadavky o notifikaci na server. V případě nějaké události server odešle upozornění a uživatel tak může být varován. Tento systém zahrnuje dvě role odběratel a oznamovatel. Odběratel reprezentuje uživatele, který odebírá varovné hlášení o mimořádných událostech. Oznamovatel je server, který odesílá upozornění na základě žádosti od odběratele.

V článku CAP-ONES: an emergency notification system for all [6] se autoři zabývají návrhu systému založenému na znalostní bázi pro upozornění na nebezpečí. Pro upozornění na různá zařízení používají EMS (Emergency Management Systems)[7], který komunikuje s uživatelem. Tento systém je založen na dvou charakteristikách. První je znalostní báze o dostupnosti, zařízeních, zdravotních postiženích, mimořádných událostí a médií, takže výstraha může být přizpůsobena podle mnoha parametrů. Druhou je protokol CAP (Common Alerting Protocol)[8], který zprostředkovává komunikaci mezi tímto a již existujícími systémy.

Další možnost se jak upozornit uživatele na mimořádné události je popsána v článku Personalized Alert Notifications and Evacuation Routes in Indoor Environments[9], který navazuje na řešení popsané v předchozím článku. Zde je použita architektura zahrnující tři prvky, oproti předchozí

kde byly navrženy dva. Prvním je tzn. CAP-ONES ten má za úkol upozornit na mimořádné události. Dalším článkem je NERES, který definuje evakuační plán a vytváří uživateli evakuační trasy. Posledním prvkem je rozhraní iNERES, ten dostane zprávu NERES a zobrazí evakuační plán na telefonu.

Tato popsání řešení však úplně nevystihují náš problém, kterým je získání informací o mimořádných událostech z neplaceného zdroje. Tyto informace poté jednoduše a přehledně zobrazit uživateli. Výše popsaná řešení se zaměřují na to, jak uživateli v případě nouze ukázat nouzový východ z budovy, nebo mu popíše, jakým způsobem by se měl v této situaci zachovat. Naším cílem je na tyto události pouze upozornit. Mělo by se jednat o výstrahu předanou uživateli. Všechny tyto informace jsou samozřejmě dostupné i na internetu pomocí webových prohlížečů, ale výhoda mobilních telefonů je v tom, že je máme stále u sebe a tak může být uživatel informován okamžitě při vzniku mimořádné události. V následující kapitole bude detailněji popsáno nové řešení.

## III. NEW SOLUTION / NOVÉ ŘEŠENÍ

V předešlé kapitole byl definován problém, který bude zde zkoumán, a detailně popsán.

Naším cílem je tedy vytvořit aplikaci pro mobilní telefon, která bude přijímat informace o mimořádných událostech z nějakého volně dostupného zdroje a tyto informace zobrazí uživateli v podobě upozornění.

V současné době je na trhu celá řada OS. Hlavními zástupci, jsou ale pouze tři. Android s podílem na trhu 84.4%, iOS (11.7%) a Windows Phone 8 (2.9%)[10]. Ostatní operační systémy jsou pouhou modifikací Androidu nebo mají na trhu jen minimální podíl.

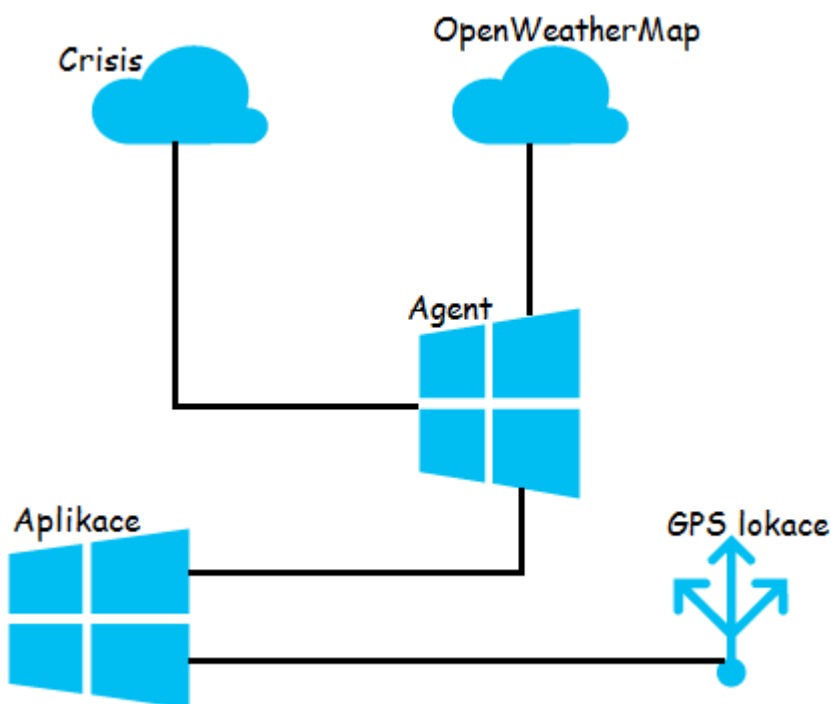
Jako platforma pro mobilní aplikaci byl zvolen Windows Phone 8, dále jen jako Windows Phone. Windows Phone je mobilní operační systém vyvíjený firmou Microsoft. Nahradil svého předchůdce Windows Mobile v únoru roku 2010. Windows Phone je využíván v různých telefonech od firem HTC, Dell, Samsung a LG. Na začátku roku 2011 bylo oznámeno, že systém Windows Phone bude využíván jako primární operační systém v telefonech Nokia.

Aplikace se vyvíjejí pomocí platformy Silverlight [11] a XNA frameworku [12]. Obě tyto technologie jsou postavené nad .NET Frameworkem a programovat se dá pomocí jazyků C#, Visual Basic .NET a F#. Jako programovacím prostředím je nejvhodnější Visual Studio, to je v základní verzi zdarma. Jedinou podmínkou je použití OS Windows Vista, Windows 7, 8, 8.1. Místo, kde lze stahovat nebo kupovat aplikace pro Windows Phone, se nazývá Windows Store. Pokud chceme publikovat vlastní aplikaci, je nutné se zaregistrovat jako vývojář na webu Microsoftu a zaplatit roční poplatek 99 dolarů. Pro studenty je registrace zdarma v rámci programu DreamSpark. Hotovou aplikaci je dále potřeba odeslat do certifikačního procesu. Tam je aplikace ověřována Microsoftem pokud splňuje podmínky pro publikování je certifikována, podepsána a může být zveřejněna na Windows Store. Podmínky na aplikaci jsou poměrně přísné, hlídají jak obsah (používání vulgárních výrazů, ochranných známek apod.), tak technické řešení (využití zdrojů, jazyková lokalizace apod.).

Tímto byla představena technologie, nyní k popisu problému, získávání a informování o mimořádných událostech. Najít nějaký volně dostupný zdroj informací o mimořádných událostech bylo dosti obtížné, většina z nich je placená a nejsou dobře aktualizované pro Českou Republiku. Zdrojem informací byly tedy vybrány dva zdroje. Prvním je Crisis [13], tento web poskytuje informace o všech mimořádných událostech, jediný problém je, že dat pro Českou Republiku obsahuje velice málo, průměrně je to jedna událost za tři měsíce. Jako druhým zdrojem je OpenWeatherMap[14], to poskytuje rychlé a přehledné informace o počasí na celém světě. Tyto informace jsou

předávány do pomoci jednoduchého API, a pro identifikaci klienta stačí mít klíč, který získá každý při registraci.

Aplikace se bude skládat ze dvou částí, agent a vlastní aplikace, ta bude hodně podobná klasickému organizátoru nebo budíku. Uživatel si nastaví čas, a zda chce být upozorněn v případě mimořádné události. Poté co si uživatel nastaví preferované funkce, přichází na řadu druhá část aplikace Agent. Ten bude na pozadí sbírat a vyhodnocovat informace z obou výše zmíněných zdrojů a na základě těchto informací informovat uživatele. Agentem sbírané informace jsou vždy z místa, kde se právě uživatel nachází, místo je lokalizováno pomocí GPS, pokud je k dispozici nebo pomocí Wifi.



Obr. 1. Schéma navrhované aplikace

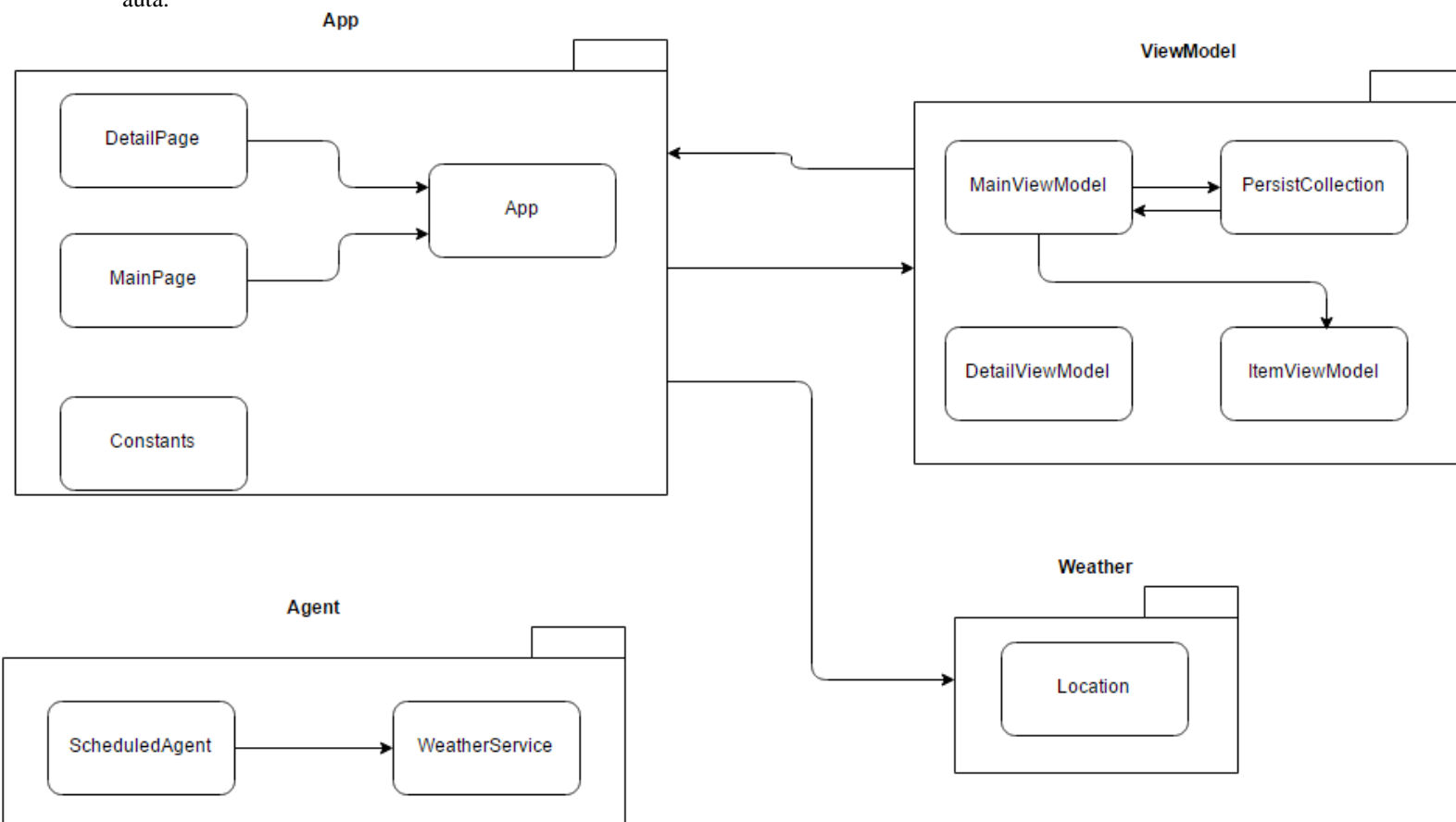
#### IV. IMPLEMENTATION / IMPLEMENTACE ŘEŠENÍ

Předchozí kapitola pojednává o novém námi definovaném řešení, v této kapitole bude rozebrána implementace tohoto řešení.

Implementace je formou mobilní aplikace a platformě Windows Phone 8. Aplikace bude mít podobnou funkci jako organizér nebo budík, s tím rozdílem, že bude sledovat teplotu v noci nebo v uživatelem zvolený čas a na základě teploty a vlhkosti vzduchu vyhodnotí, zda je pravděpodobný vznik námrazy. Podle této informace upraví čas upozornění. Ušetří tak uživateli starost například s tím aby předtím než jde spát, kontroloval počasí a pokud má být v noci námraza nastavil si budík o chvíli dříve a mohl si tak očistit skla u auta.

Uživatel si jednoduše zvolí čas buzení nebo upozornění, a zda chce mít k dispozici funkci kontroly mimořádných událostí (v tomto případě vzniku námrazy). Aplikace poté automaticky naplánuje agenta, který bude hlídat teplotu a vlhkost vzduchu, pokud hodinu před naplánovanou aktivitou budou podmínky vyhovovat vzniku námrazy, agent automaticky posune čas aktivity o 20 minut dříve.

Celé schéma vytvořené aplikace je zobrazeno na obrázku č. 2. Aplikace obsahuje dva moduly Agent a CoreApp. Agent je nezávislý na aplikaci a se stará o kontrolu a stahování dat z webové služby. Samotná aplikace obsahuje uživatelské prostředí, logiku pro upozornění uživatele a komponentu, která se stará o lokalizaci uživatele.



Obr. 2. Schéma implementované aplikace

Jelikož se aplikace připojuje k webové službě, je jako první krok je potřeba se zaregistrovat na OpenWeatherMap[14]. Pro registraci stačí vyplnit jednoduchý formulář a na konci získáme požadovaný klíč. Tento klíč nám bude sloužit jako identifikace pro připojení ke službám.

Ve druhé kapitole bylo popsáno, že aplikace se bude připojovat také ke službě Crisis, tato služba byla ovšem vyřazena z aplikace, protože neposkytuje dobře aktualizovaná data. Pro ČR obsahuje pouze jednu mimořádnou událost za tři měsíce a pouze pro Prahu. Z toho důvodu je pro naši aplikaci nevhodná a je použita pouze služba OpenWeatherMap.

Aplikace je tvořena strukturou MVVM (Model View ViewModel)[15]. Model-View-ViewModel je návrhový vzor pro WPF[16] aplikace. Nabízí řešení, jak oddělit logiku aplikace od uživatelského rozhraní. Vše je potom přehlednější a případné změny jsou implementačně snazší. MVVM odděluje data, stav aplikace a uživatelské rozhraní a tím zaručuje znovupoužitelnost.

Uživatelské rozhraní aplikace bylo navrženo tak aby bylo co nejjednodušší. Úvodní strana, na kterou se uživatel dostane po spuštění aplikace, obsahuje seznam nastavených událostí, zda je tato událost aktivní a zda má kontrolu mimořádných událostí. Ve spodní části obrazovky je tlačítko na přidání nové události. V obrazovce pro přidávání nových událostí najdeme kolonku pro zadání názvu události, výběr času události a tlačítko pro zapnutí nebo vypnutí kontroly mimořádných událostí. Po potvrzení vytvořené události, se automaticky vytvoří i agent, který bude sledovat a stahovat informace z webové služby. U každé události lze na úvodní obrazovce aktivovat nebo zrušit naplánovanou událost popřípadě upozornění na mimořádné události.

Ukázková aplikace by měla správně fungovat na jakémkoli zařízení, které má operační systém Windows Phone 8. Je to z toho důvodu, že Microsoft stanovil standardy pro zařízení, na které je instalován tento operační systém. Každé tak musí splňovat podmínky: min. 512MB RAM, GPS, NFC, Wifi a dvou jádrový procesor. To vývojáři poskytuje značnou výhodu oproti Androidu, který má velikou škálu zařízení a vyvíjet aplikace, tak aby se zobrazovaly správně na všech zařízeních, je veliký problém.

Při vývoji aplikace se vyskytl problém s komunikací mezi oběma moduly, Agentem a CoreApp. Aplikaci využívá pro ukládání dat svoji izolovanou paměť. Nastává ovšem problém pokud aplikace běží ve více vláknech a je potřeba, aby obě vlákna viděli ve zmíněné paměti to samé. V našem případě nastal problém, když agent uložil informaci o počasí do paměti a aplikace o této informaci nevěděla. Řešení tohoto problému byla synchronizace procesu ukládání dat do paměti. Toho bylo dosaženo pomocí třídy Mutex.

Tato kapitola obsahovala popis konkrétní implementace ukázkové aplikace. Další kapitola bude věnována testování aplikace.

## V. TESTING OF DEVELOPED APPLICATION / TESTOVÁNÍ VYVINUTÉ APLIKACE - ŘEŠENÍ

Implementovanou aplikaci je nutné otestovat. Z testu by mělo vidět, zda námi navržené řešení bylo v nějakém směru

lepší. Aplikaci lze otestovat na základě predikce mimořádné události, v našem případě na vznik námrazy. Dalším testem je úspěšnost lokalizace uživatele.

### Test predikce vzniku námrazy.

Byl proveden test o 10 měření ve třech různých lokalitách a se třemi různými nastaveními. Jak je vidět z tabulky Tab. 1. nejlepší kombinace, která zaručuje největší četnost výskytu námrazy je s teplotním rozmezí 0 / -4 °C a vlhkostí vzduchu 70 %.

Lokalita	Teplotní rozmezí [°C]	Vlhkost vzduchu [%]	Četnost výskytu námrazy
Lanškroun	0 / -4	70	8
Lanškroun	-1 / -4	75	7
Lanškroun	0 / -5	80	8
Hradec Králové	0 / -4	70	8
Hradec Králové	-1 / -4	75	6
Hradec Králové	0 / -5	80	9
Opatovice nad Labem	0 / -4	70	10
Opatovice nad Labem	-1 / -4	75	9
Opatovice nad Labem	0 / -5	80	9

Tab. 1. Test predikce námrazy

V testu může hrát roli i lokalita měření, z toho důvodu že na vesnicích (Opatovice nad Labem) je teplota obecně nižší než ve městech (Hradec Králové, Lanškroun) kde je více lidí, více aut a více smogu.

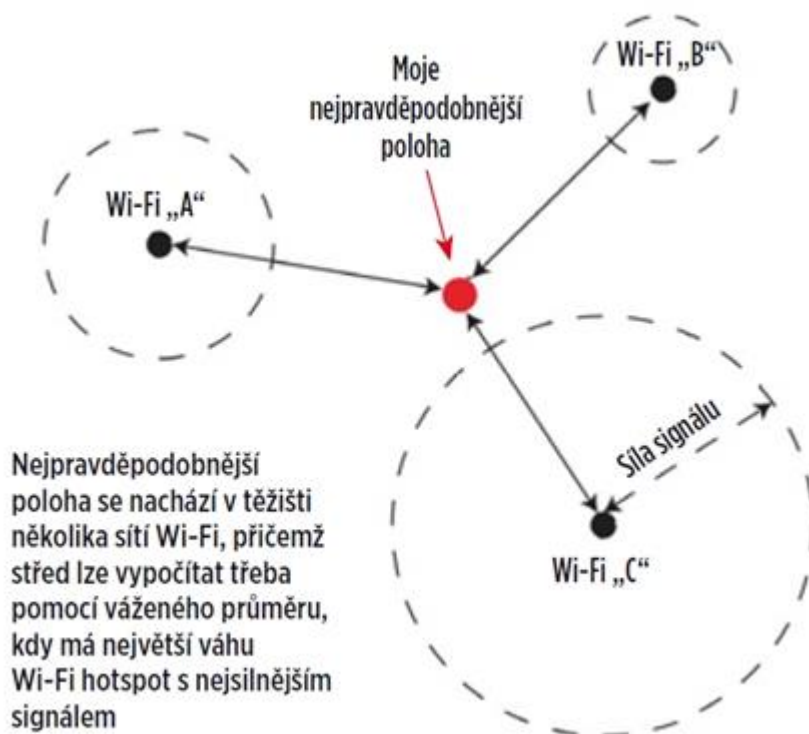
### Test rychlosti a úspěšnosti lokalizace

Druhý test bude testovat úspěšnost a rychlost lokalizace uživatele. Protože aplikace má k dispozici dvě možnosti jak uživatele lokalizovat, pomocí Wifi nebo GPS. Test bude sledovat rychlost a správnost nalezené lokality.

Lokalita	Druh připojení	Rychlost [s]	Úspěšnost [%]
Lanškroun	Wifi	0.5	100
Lanškroun	GPS	1.2	100
Hradec Králové	Wifi	0.5	100
Hradec Králové	GPS	1.0	100
Opatovice nad Labem	Wifi	0.75	60
Opatovice nad Labem	GPS	2	100

Tab. 2. Test rychlosti a úspěšnosti lokalizace

Test byl prováděn opět na deseti pokusech, přičemž se ukázalo, že GPS je sice pomalejší, ale poskytuje stoprocentní úspěšnost. To v případě vesnic je důležité vodítko. Jak je vidět z tabulky Tab. 2. úspěšnost Wifi v Opatovicích nad Labem je pouze 60% to je z toho důvodu, že vesnice se nachází blízko Hradce Králové (3km). Wifi získává polohu na základě IP adresy a dostupných WiFi sítí. V případě požadavku na lokalizaci se sesbírají údaje o okolních WiFi sítích, tedy jejich názvy, MAC adresy routerů a sílu signálu. Tyto údaje pošle lokalizační službě (Google), která je porovná se svou databází WiFi routerů se známým umístěním a na základě toho je vypočítána pravděpodobná poloha, která se pošle zpět [17]. Při výpočtu používá také IP adresu, z níž požadavek přišel obr. 3.



Obr. 3. Schéma zjišťování pravděpodobné pozice pomocí Wi-Fi [17].

Tato metoda je nepřesná právě kvůli nemožnosti získání přesné pozice ale pouze odhadu této pozice. Ve městech tato metoda lokalizace funguje správně kvůli jejich velké rozloze. Odhad pozice uživatele tedy vždy připadne do tohoto města. Oproti ve vesnici kde je rozloha malá. Na základě testu bych doporučil mít GPS vždy zapnutou až už je nějaká Wifi síť dostupná nebo ne. Aplikaci si vždy preferuje GPS lokaci před Wifi, hlavně kvůli přesnosti.

Další měřené kritérium v testu byla rychlost, jakou se poloha získává. V tomto kritériu jednoznačně zvítězila Wifi. Jednak nemusí hledat pozici družice a také polohu pouze odhaduje.

Z výsledků obou testů, lze odvodit ideální nastavení aplikace. Tedy teplotní rozmezí mezi 0 / -4 °C, vlhkost vzduchu 70% a zapnuté GPS. Takto nastavená aplikace by měla zaručovat vysokou pravděpodobnost úspěšné predikce námrazy.

V porovnání s ostatními řešeními, které jsou popsány v kapitole dvě. Tato aplikace sice neposkytuje plán uniku, ale dokáže spolehlivě upozornit na mimořádnou událost v podobě námrazy. Naším cílem také bylo najít nějaký zdroj, který by poskytoval informace o mimořádných událostech a byl zdarma. Tento cíl se nepodařilo uskutečnit, jediným takovým to zástupcem byl Crisis a ten je bohužel špatně aktualizovaný.

## VI. CONCLUSIONS / ZÁVĚRY

Na začátku byl definovaný cíl vytvoření aplikace, která by informovala uživatele o mimořádných událostech pomocí notifikací na mobilní telefon. Zdroj informací o událostech musí být zdarma, aby mohlo být dosaženo nulové ceny aplikace. Tento cíl se podařilo splnit pro naše území pouze částečně. Aplikace je funkční a spolehlivá, ale takový zdroj informací, který by byl zdarma a poskytoval aktualizovaná data o mimořádných událostech, se najít podařilo jen částečně. Byl použit OpenWeatherMap jako zástupce informací o počasí. Zdroj Crisis jako zástupce ostatních mimořádných událostí je pro Českou Republiku bohužel špatně aktualizovaný. To ovšem neznamená, že v jiných větších státech by tento zdroj informací fungoval spolehlivě stejně jako v našem případě zdroj OpenWeatherMap.

## REFERENCES / REFERENCE

- [1] ABOWD, Gregory D, Barry BRUMITT a Steven SHAFER. Ubicomp 2001: ubiquitous computing : International Conference, Atlanta, Georgia, USA, September 30-October 2, 2001 : proceedings. New York: Springer, c2001, xiii, 372 p. ISBN 35-404-2614-0.
- [2] KITCHENS, Fred L., Josh M. MCGUIRE a Sushil K. SHARMA. Emergency preparation and mobile notification through cluster computing. International Journal of Mobile Communications. Chichester: Wiley, 2006, Volume 4, Number 1/2006, s. 82-96.
- [3] LIBERTY, Jesse, Philip JAPIKSE a Jon GALLOWAY. Notifications. Pro Windows 8.1 Development with XAML and C#. Berkeley, CA: Apress, 2014, s. 209. DOI: 10.1007/978-1-4302-4048-8\_9. Dostupné z: [http://link.springer.com/10.1007/978-1-4302-4048-8\\_9](http://link.springer.com/10.1007/978-1-4302-4048-8_9)
- [4] LEE, Henry a Eugene CHUVYROV. Push Notifications. Beginning Windows Phone App Development. Berkeley, CA: Apress, 2012, s. 405. DOI: 10.1007/978-1-4302-4135-5\_17. Dostupné z: [http://link.springer.com/10.1007/978-1-4302-4135-5\\_17](http://link.springer.com/10.1007/978-1-4302-4135-5_17)
- [5] ISHII, Naohiro a Roger LEE. 5th IEEE/ACIS International Conference on Computer and Information Science: (ICIS 2006) in conjunction with 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse (COMSAR 2006) : proceedings : 10-12 July 2006, Honolulu, Hawaii. Los Alamitos, Calif.: IEEE Computer Society, c2006. ISBN 9780769526133.
- [6] MALIZIA, Alessio, Pablo ACUNA, Teresa ONORATI, Paloma DIAZ a Ignacio AEDO. CAP-ONES: an emergency notification system for all. International Journal of Emergency Management. 2009, vol. 6, 3/4, s. 302-. DOI: 10.1504/IJEM.2009.031568. Dostupné z: <http://www.inderscience.com/link.php?id=31568>
- [7] HERNÁNDEZ, Josefa Z. a Juan M. SERRANO. Knowledge-based models for emergency management systems. Expert Systems with Applications. 2001, vol. 20, issue 2, s. 173-186. DOI: 10.1016/S0957-4174(00)00057-9. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S0957417400000579>
- [8] SAINT-ANDRE, Peter a Boyd FLETCHER. Common Alerting Protocol (CAP) Over XMPP [online]. 1999 - 2014 [cit. 2014-12-19]. Dostupné z: <http://xmpp.org/extensions/xep-0127.html#appendix-legal>
- [9] AEDO, Ignacio, Shuxin YU, Paloma DÍAZ, Pablo ACUÑA a Teresa ONORATI. Personalized Alert Notifications and Evacuation Routes in Indoor Environments. Sensors. 2012, vol. 12, issue 12, s. 7804-7827. DOI: 10.3390/s120607804. Dostupné z: <http://www.mdpi.com/1424-8220/12/6/7804/>
- [10] Smartphone OS Market Share, Q3 2014. IDC Analyze the future [online]. 2014 [cit. 2014-12-06]. Dostupné z: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>
- [11] Microsoft. Microsoft Silverlight [online]. 2014 [cit. 06.12.2014]. Dostupné z: <http://www.microsoft.com/cze/web/silverlight/default.aspx>
- [12] Microsoft. XNA Developer Center [online]. 2014 [cit. 06.12.2014]. Dostupné z: <http://msdn.microsoft.com/cs-cz/xna>
- [13] Crisis: The firehose of global crisis data [online]. 2014 [cit. 2014-12-19]. Dostupné z: <http://crisis.net/>
- [14] OpenWeatherMap [online]. 2014 [cit. 2014-12-19]. Dostupné z: <http://openweathermap.org/>
- [15] Implementing the Model-View-ViewModel Pattern. Microsoft Developer Network [online]. 2010 [cit. 2014-12-23]. Dostupné z: <http://msdn.microsoft.com/en-us/library/ff798384.aspx>
- [16] Introduction to WPF. Microsoft Developer Network [online]. 2010 [cit. 2014-12-23]. Dostupné z: [http://msdn.microsoft.com/en-us/library/aa970268\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/aa970268(v=vs.110).aspx)
- [17] Jak je možné, že mobil ví, kde zrovna jsme? Více na: [http://www.zive.cz/clanky/jak-je-mozne-ze-mobil-vi-kde-zrovna-jsme/lokalizace-pomoci-wi-fi/sc-3-a-174857-ch-93700/default.aspx?utm\\_medium=selfpromo&utm\\_source=zive&utm\\_campaign=copylink](http://www.zive.cz/clanky/jak-je-mozne-ze-mobil-vi-kde-zrovna-jsme/lokalizace-pomoci-wi-fi/sc-3-a-174857-ch-93700/default.aspx?utm_medium=selfpromo&utm_source=zive&utm_campaign=copylink). Zive.cz [online]. 2014 [cit. 2015-01-08]. Dostupné z: <http://www.zive.cz/clanky/jak-je-mozne-ze-mobil-vi-kde-zrovna-jsme/lokalizace-pomoci-wi-fi/sc-3-a-174857-ch-93700/default.aspx>